

Programming Assignment 1

Due: 11:59pm, Saturday, January 9

Overview

The goals of this section are to:

1. Get your PC set up to access the remote ieng6.ucsd.edu servers
 - a. Set up a ssh client, file transfers, x11 forwarding
2. Introduction to Linux commands
3. Vim/Emacs editors
4. Compiling and debugging your first Java programs

1. Desktop setup

Complete the instructions at <https://sites.google.com/a/eng.ucsd.edu/cse-11-winter-2015-mwf/remote-lab-access>.

Also, set up X11 forwarding:

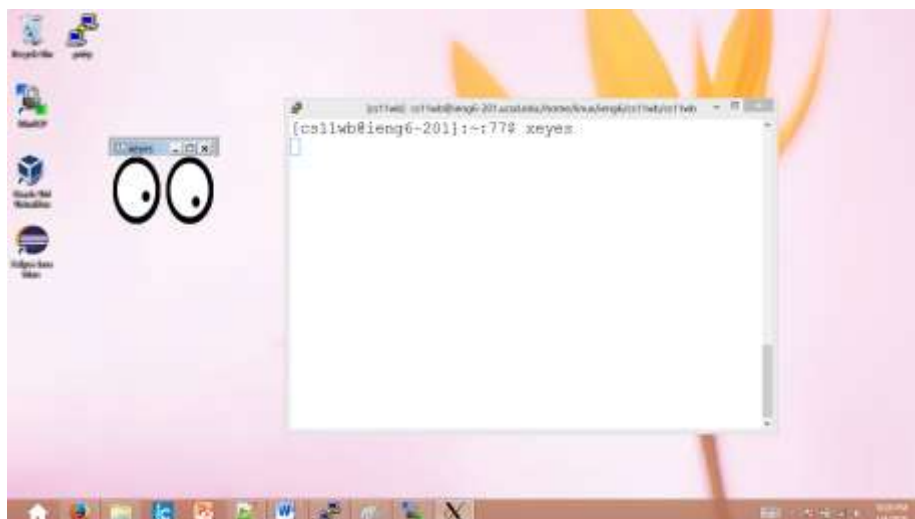
(Windows):

<https://wiki.utdallas.edu/wiki/display/FAQ/X11+Forwarding+using+Xming+and+PuTTY>.

(OS X): <https://support.apple.com/en-us/HT201341>. Once installed, you'll need to re-login to the system with the command:

```
$ ssh -X cs11wb??@ieng6.ucsd.edu
```

Once done with the above steps, take a screenshot (google how to do this if unsure) of your PC connected to the ieng6.ucd.edu server after executing the command **xeyes**. If everything is set up correctly, it should look similar to:



Later in the document you'll upload this image to the ieng6 servers for your homework submission.

2. Basic Linux commands

Complete tutorials 1-5 at http://cseweb.ucsd.edu/classes/wi16/cse11-b/hw/linux_tutorial/index.html

At the end of the tutorial, you will have created a few files and directories. We will be checking the correctness of the files.

Also, at this time you can now upload your screenshot taken in Part 1 of the assignment. Name the file screenshot.xxx (where xxx is the file type, e.g. png, jpg, etc) and copy to your HW1 directory on the ieng6 servers. If you're on windows, you'll use a program like WinSCP. For those of you on Macs, you'd use the command SCP. Details were given in part 1 of the assignment.

3. Editors

There are 2 big editors in Linux, emacs and vi(m). You'll need to get comfortable using one or the other. We will not be using IDEs (e.g. eclipse) in this class.

Grab the setup files for emacs and vim from /home/linux/ieng6/cs11wb/public/HW1 and place in your home directory:

```
$ cp /home/linux/ieng6/cs11wb/public/HW1/.emacs ~/
$ cp /home/linux/ieng6/cs11wb/public/HW1/.vimrc ~/
```

These 2 files will help your program conform to the required program styles used in this course. Note that they go in your home directory. When emacs/vim start, they'll load these settings files, but only if they're in the correct location. Next, try to create and edit a text file using emacs:

```
$ emacs test.txt
```

This will open emacs in a new window. Another option is to open it within the terminal:

```
$ emacs -nw test.txt
```

Just start typing text. See http://ocean.stanford.edu/research/quick_emacs.html for basic commands on how to save, cut, copy, search, page-up, page-down, and exit emacs.

Next, do the same test with vim:

```
$ vim test.txt
```

You can learn basic commands for vim via the command line tool, vimtutor.

```
$ vimtutor
```

4. Writing, compiling, and debugging your first Java program

Using either vim or emacs, type out the following program in the file ~/HW1/Hello.java

```
/**
 * @author: <fill this in>
 * @email: <fill this in>
 * @pid: <fill this in>
 * @about: This program prints Hello World
 *
 * @compile: javac Hello.java
 * @run: java Hello
 *        Hello World
 */
public class Hello {

    public static void main(String[] args) {
        System.out.println("Hello World");
    }

}
```

Compile and execute your program:

```
$ javac Hello.java
$ java Hello
```

Next, execute the following command

```
$ cp /home/linux/ieng6/cs11wb/public/HW1/Debug.java ~/HW1/
```

If you attempt to compile the program Debug.java via:

```
$ javac Debug.java
```

You'll get a few programming errors. Open the file Debug.java with emacs or vim and find and fix each of the errors. For each error, comment out the original line and replace with the corrected line. Make sure your program now compiles and runs via:

```
$ javac Debug.java
$ java Debug
```

```
Enter your name: Adam Jundt
Your name is: Adam Jundt
```

Also, for each .java file be sure to include your name, student ID, and username.

Turnin Instructions

Remember the deadline to turn in your assignment is Saturday, January 9 by 11:59pm.
When you are ready to turn in your program in, type in the following command and answer the prompted questions:

```
$ cd ~/
$ bundleP1
Good; all required files are present:

    HW1

Do you want to go ahead and turnin these files? [y/n]y
OK.  Proceeding.

Performing turnin of approx. 6144 bytes (+/- 10%)
Copying to /home/linux/ieng6/cs11wb/turnin.dest/cs11wb.P1
...
Done.
Total bytes written: 31744
Please check to be sure that's reasonable.
Turnin successful.
```

bundleP1 will look for the directory HW1 and will compress the files into a single turnin submission. You can turnin your program multiple times. The bundle program will ask you if you want to overwrite a previously-turned in homework. ONLY THE LAST SUBMISSION IS USED!